BOOTS

STARDDRIV

LIS

STARDDRIV
V04-000

N 8
- Driver read routines for STASYSGEN, ST 16-SEP-1984 00:04:59   VAX/VMS Macro V04-00   Page 1
                                        4-SEP-1984 23:06:17 [BOOTS.SRC]STARDDRIV.MAR;1   (1)

```
0000    1                    .TITLE  STARDDRIV - Driver read routines for STASYSGEN, STACONFIG
0000    2                    .IDENT  'V04-000'
0000    3
0000    4          ;
0000    5          ;*******************************************************************************
0000    6          ;*                                                                             *
0000    7          ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                    *
0000    8          ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                     *
0000    9          ;*  ALL RIGHTS RESERVED.                                                       *
0000   10          ;*                                                                             *
0000   11          ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED      *
0000   12          ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE      *
0000   13          ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER      *
0000   14          ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY      *
0000   15          ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY      *
0000   16          ;*  TRANSFERRED.                                                               *
0000   17          ;*                                                                             *
0000   18          ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE      *
0000   19          ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT      *
0000   20          ;*  CORPORATION.                                                               *
0000   21          ;*                                                                             *
0000   22          ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS      *
0000   23          ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                    *
0000   24          ;*                                                                             *
0000   25          ;*                                                                             *
0000   26          ;*******************************************************************************
0000   27          ;
0000   28
0000   29          ;++
0000   30          ; FACILITY:      STANDALONE SYSGEN, STANDALONE CONFIGURE
0000   31          ;
0000   32          ; ABSTRACT:      READS DRIVERS INTO MEMORY.
0000   33          ;
0000   34          ; ENVIRONMENT: USER, EXEC, AND KERNEL MODE
0000   35          ;
0000   36          ; AUTHOR: STEVE BECKHARDT,     CREATION DATE:  18-SEP-1979
0000   37          ;
0000   38          ; MODIFIED BY:
0000   39          ;
0000   40          ;     V03-003 KDM0090          Kathleen D. Morse      10-Nov-1983
0000   41          ;             Make file header buffer fall on word boundry instead
0000   42          ;             of a byte boundry, so that STASYSGEN works on MicroVAX I.
0000   43          ;
0000   44          ;     V03-002 WMC0001          Wayne Cardoza          09-Aug-1983
0000   45          ;             Make it handle non-contiguous files.
0000   46          ;
0000   47          ;     V03-001 MSH0001          Maryann Hinden         07-Jul-1983
0000   48          ;             Move to separate module.
0000   49          ;--
```

```
                    0000      51              .SBTTL  DECLARATIONS
                    0000      52
                    0000      53  ; INCLUDE FILES:
                    0000      54  ;
                    0000      55
                    0000      56  ;
                    0000      57  ; MACROS:
                    0000      58  ;
                    0000      59
                    0000      60  ;
                    0000      61  ; EQUATED SYMBOLS:
                    0000      62  ;
                    0000      63
        0000000D    0000      64  CR = ^XD                                    ; ASCII carriage return
        0000000A    0000      65  LF = ^XA                                    ; ASCII line feed
                    0000      66
                    0000      67  ;
                    0000      68  ; OWN STORAGE:
                    0000      69  ;
                    0000      70
        00000000    0000      71          .PSECT  BOO$SYSGEN,WRT,WORD
                    0000      72
        00000200    0000      73  FILHDR: .BLKB   512                         ; Buffer for file header
                    0200      74
        00000400    0200      75  IXFHDR: .BLKB   512                         ; Buffer for index file header
                    0400      76
                    0400      77  RTRVPTR:                                    ; Buffer for 128 retrieval pointer
        00000600    0400      78          .BLKB   512
                    0600      79
                    0600      80  RTRVLEN:                                    ; Length of retreival pointers
        00000604    0600      81          .BLKL   1
                    0604      82
                    0604      83  RTRVDESC:                                   ; Descriptor for retrieval pointer buffer
        00000200    0604      84          .LONG   512
        00000400'   0608      85          .ADDRESS RTRVPTR
                    060C      86
                    060C      87  STATBLK:                                    ; Statistics block
        00000610    060C      88          .BLKL   1                           ; LBN of first block of file
        00000614    0610      89          .BLKL   1                           ; Size of file in blocks
                    0614      90
                    0614      91  IOSTBLK:                                    ; I/O status block
        0000061C    0614      92          .BLKL   2
                    061C      93
                    061C      94  FILNAM_DSC:                                 ; File name descriptor
        00000000    061C      95          .LONG   0                           ; Size (in bytes)
        00000625'   0620      96          .ADDRESS FILNAMBFR                  ; Address of file name
                    0624      97
                    0624      98  FILNAMBFRLF:                                ; Line feed before file name buffer
        0A          0624      99          .ASCII  <LF>
                    0625     100  FILNAMBFR:                                  ; File name buffer
        00000635    0625     101          .BLKB   16
                    0635     102
```

```
                        0635    104                    .SBTTL  IOGEN$READDRIV - Read in Driver
                        0635    105  ;++
                        0635    106  ; FUNCTIONAL DESCRIPTION:
                        0635    107  ;
                        0635    108  ;       This routine reads the driver into memory by opening it
                        0635    109  ;       with FIL$OPENFILE and reading it with a QIO.
                        0635    110  ;
                        0635    111  ; CALLING SEQUENCE:
                        0635    112  ;
                        0635    113  ;       BSBW    IOGEN$READDRIV
                        0635    114  ;
                        0635    115  ; INPUT PARAMETERS:
                        0635    116  ;
                        0635    117  ;       R0      Address of filename counted string
                        0635    118  ;       R3      Address of location to store channel number
                        0635    119  ;       R4      Address of two longword array to return address range
                        0635    120  ;               created by $EXPREG.
                        0635    121  ;
                        0635    122  ; IMPLICIT INPUTS:
                        0635    123  ;
                        0635    124  ;       None
                        0635    125  ;
                        0635    126  ; OUTPUT PARAMETERS:
                        0635    127  ;
                        0635    128  ;       R0      Completion code
                        0635    129  ;
                        0635    130  ; IMPLICIT OUTPUTS:
                        0635    131  ;
                        0635    132  ;       None
                        0635    133  ;
                        0635    134  ; COMPLETION CODES
                        0635    135  ;
                        0635    136  ;       Those returned by FIL$OPENFILE, $EXPREG, and $QIO
                        0635    137  ;
                        0635    138  ; SIDE EFFECTS:
                        0635    139  ;
                        0635    140  ;       R0 - R2 are used as scratch registers
                        0635    141  ;
                        0635    142  ;--
                        0635    143
                        0635    144  IOGEN$READDRIV::
            51  80  9A  0635    145          MOVZBL  (R0)+,R1                ; Get length of filename
            09  51  D1  0638    146          CMPL    R1,#9                   ; Longer than 9 characters?
                06  15  063B    147          BLEQ    10$                     ; No
        50  0000'8F  3C  063D   148          MOVZWL  #SS$_BADFILENAME,R0     ; Yes, error
                    05  0642    149          RSB
                        0643    150
   D4 AF  51  04  C1   0643    151  10$:     ADDL3   #4,R1,FILNAM_DSC        ; Store filename size + 4
            3C  BB     0648    152          PUSHR   #^M<R2,R3,R4,R5>        ; Save registers
   D6 AF  60  51  28   064A    153          MOVC3   R1,(R0),FILNAMBFR       ; Copy filename into local buffer
  63  4558452E 8F  D0  064F    154          MOVL    #^A/.EXE/,(R3)          ; Append filetype
            3C  BA     0656    155          POPR    #^M<R2,R3,R4,R5>        ; Restore registers
                        0658    156
                        0658    157          ;
                        0658    158          ; Open the file
                        0658    159          ;
                        0658    160
```

STARDDRIV
V04-000

D 9
- Driver read routines for STASYSGEN, ST 16-SEP-1984 00:04:59   VAX/VMS Macro V04-00   Page 4
IOGEN$READDRIV - Read in Driver   4-SEP-1984 23:06:17 [BOOTS.SRC]STARDDRIV.MAR;1   (3)

```
              A9 AF   7F 0658   161          PUSHAQ  RTRVDESC          ; Buffer for retrieval pointers
              A2 AF   9F 065B   162          PUSHAB  RTRVLEN           ; Get length of retrieval pointer buffer use
        52    AB AF   7E 065E   163          MOVAQ   STATBLK,R2        ; Get address of statistics block
                 62   7F 0662   164          PUSHAQ  (R2)              ; Push address of statistics block
            F998 CF   9F 0664   165          PUSHAB  FILHDR            ; Push address of file header buffer
            FB94 CF   9F 066B   166          PUSHAB  IXFHDR            ; Push address of index file hdr bfr
              AD AF   7F 066C   167          PUSHAQ  FILNAM_DSC        ; Push address of filename descriptor
                 63   3F 066F   168          PUSHAW  (R3)              ; Push address of loc. to store channel
       00000000'EF   07 FB 0671 169          CALLS   #7,FIL$OPENFILE   ; Open the file
              0E 50   E9 0678   170          BLBC    R0,20$            ; Error
        84 AF  82 AF  D1 067B   171          CMPL    RTRVLEN,RTRVDESC  ; Did we overflow buffer
                 26   19 0680   172          BLSS    40$               ; No
        50 00000000'EF  D0 0682 173          MOVL    SS$_FILNOTCNTG,R0
              F974'  30 0689   174 20$:      BSBW    PUTERROR          ; Output error message
        50  8D AF  D0 068C   175          MOVL    FILNAM_DSC,R0     ; Get length of filename
        90 AF40  0D  90 0690   176          MOVB    #CR,FI[NAMBFR[R0] ; Put CR at end of buffer
           8D AF40  94 0695   177          CLRB    FILNAMBFR+1[R0]   ; Make it ASCIZ
                 7E   7C 0699   178          CLRQ    -(SP)             ; No input buffer
              86 AF   9F 069B   179          PUSHAB  FILNAMBFRLF       ; Push address of filename
       00000000'EF   03 FB 069E 180          CALLS   #3,BOO$READPROMPT ; Output driver name
                 50   D4 06A5   181          CLRL    R0                ; Status
                 05   06A7   182          RSB
                       06A8   183
            00E0 8F   BB 06A8   184 40$:     PUSHR   #^M<R5,R6,R7>
                       06AC   185
                       06AC   186          ; Expand the program region to create a place to read driver into
                       06AC   187          ;
                       06AC   188
                       06AC   189          $EXPREG_S       PAGCNT = 4(R2),-; # of pages
                       06AC   190                          RETADR = (R4)   ; Return address array
              4C 50   E9 06BC   191          BLBC    R0,90$                ; Error
                       06BF   192
                       06BF   193          ;
                       06BF   194          ; Read in the driver
                       06BF   195          ;
                       06BF   196
        52  FF3B CF  FD 8F 78 06BF 197      ASHL    #-3,RTRVLEN,R2    ; Number of retrieval pointers
           55  FD36 CF  9E 06C6   198       MOVAB   RTRVPTR,R5        ; Start of pointers
           56    64   D0 06CB   199          MOVL    (R4),R6           ; Start of driver buffer
           50    85   7D 06CE   200 50$:     MOVQ    (R5)+,R0          ; R0 = # of blocks, R1 = LBN,
        57 50  00000200 8F C5 06D1 201       MULL3   #512,R0,R7        ; Convert blocks to bytes
                       06D9   202          $QIOW_S CHAN = (R3),-      ; Channel number
                       06D9   203                  FUNC = #IO$_READLBLK,- ; Function
                       06D9   204                  IOSB = IOSTBLK,-   ; I/O status block
                       06D9   205                  P1 = (R6),-        ; Buffer address
                       06D9   206                  P2 = R7,-          ; Byte count
                       06D9   207                  P3 = R1           ; LBN
              0E 50   E9 06FA   208          BLBC    R0,90$            ; Error
        50  FF13 CF  3C 06FD   209          MOVZWL  IOSTBLK,R0        ; Final status
              06 50   E9 0702   210          BLBC    R0,90$            ; Error
           56    57   C0 0705   211          ADDL    R7,R6             ; New buffer
           C3 52   F5 0708   212          SOBGTR  R2,50$
                       070B   213
            00E0 8F   BA 070B   214 90$:     POPR    #^M<R5,R6,R7>
                 05   070F   215          RSB
                       0710   216
                       0710   217          .END
```

```
$$T1             = 00000001
BOO$READPROMPT     ********   X    01
CR               = 0000000D
FIL$OPENFILE       ********   X    01
FILHDR             00000000 R      01
FILNAMBFR          00000625 R      01
FILNAMBFRLF        00000624 R      01
FILNAM_DSC         0000061C R      01
IO$_READLBLK       ********   X    01
IOGEN$READDRIV     00000635 RG     01
IOSTBLK            00000614 R      01
IXFHDR             00000200 R      01
LF               = 0000000A
PUTERROR           ********   X    01
RTRVDESC           00000604 R      01
RTRVLEN            00000600 R      01
RTRVPTR            00000400 R      01
SS$_BADFILENAME    ********   X    01
SS$_FILNOTCNTG     ********   X    01
STATBLK            0000060C R      01
SYS$EXPREG         ********   GX   01
SYS$QIOW           ********   GX   01
```

```
+------------------+
! Psect synopsis !
+------------------+
```

PSECT name	Allocation	PSECT No.	Attributes										
. ABS .	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE
BOO$SYSGEN	00000710 (1808.)	01 (1.)	NOPIC	USR	CON	REL	LCL	NOSHR	EXE	RD	WRT	NOVEC	WORD

```
+----------------------------+
! Performance indicators !
+----------------------------+
```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.10	00:00:00.71
Command processing	110	00:00:00.66	00:00:02.71
Pass 1	134	00:00:01.05	00:00:03.51
Symbol table sort	0	00:00:00.01	00:00:00.01
Pass 2	54	00:00:00.44	00:00:01.00
Symbol table output	3	00:00:00.02	00:00:00.03
Psect synopsis output	1	00:00:00.01	00:00:00.01
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	333	00:00:02.30	00:00:07.98

The working set limit was 1050 pages.
4178 bytes (9 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 22 non-local and 5 local symbols.
217 source lines were read in Pass 1, producing 11 object records in Pass 2.
6 pages of virtual memory were used to define 6 macros.

F 9

STARDDRIV   - Driver read routines for STASYSGEN, ST 16-SEP-1984 00:04:59 VAX/VMS Macro V04-00   Page 6
VAX-11 Macro Run Statistics   4-SEP-1984 23:06:17 [BOOTS.SRC]STARDDRIV.MAR;1   (3)

```
                                   +------------------------------+
                                   ! Macro library statistics !
                                   +------------------------------+

Macro library name                 Macros defined
------------------                 --------------
_$255$DUA28:[BOOTS.OBJ]BOOTS.MLB;1       0
_$255$DUA28:[SYS.OBJ]LIB.MLB;1           0
_$255$DUA28:[SYSLIB]STARLET.MLB;2        6
TOTALS (all libraries)                   6
```

70 GETS were required to define 6 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:STARDDRIV/OBJ=OBJ$:STARDDRIV MSRC$:STARDDRIV/UPDATE=(ENH$:STARDDRIV)+EXECML$/LIB+LIB$:BOOTS.MLB/LIB

SYSBOOCMD
LIS

SHOWADAP
LIS

STANDCONF
LIS

SHODEV
LIS

STAPUTERR
LIS

STALOCK
LIS

STASGNMSG
LIS

STACONFIG
LIS

STASYSGEN
LIS

SYSBOOT
LIS

STARDGRIV
LIS